

Отчёт по лабораторной работе 6

**Установка и настройка системы управления базами данных
MariaDB**

Авдадаев Джамал Геланиевич

Содержание

1 Введение	5
1.1 Цель работы	5
2 Процесс работы	6
2.1 Установка MariaDB	6
2.1.1 Установка необходимых пакетов	6
2.1.2 Просмотр конфигурационных файлов	6
2.1.3 Запуск MariaDB и включение автозагрузки	7
2.1.4 Проверка работы MariaDB и прослушивания порта	8
2.1.5 Настройка безопасности MariaDB	8
2.1.6 Взаимодействие с MariaDB	9
2.2 Конфигурация кодировки символов	10
2.2.1 Просмотр статуса MariaDB	10
2.2.2 Создание конфигурации utf8	10
2.2.3 Перезапуск MariaDB и проверка	11
2.3 Создание базы данных addressbook	11
2.3.1 Создание базы и выбор её в работе	11
2.3.2 Создание таблицы city	12
2.4 Создание пользователя и настройка его прав	13
2.4.1 Создание пользователя	13
2.4.2 Назначение прав	13
2.5 Просмотр баз данных и таблиц	14
2.6 Работа с резервными копиями	15
2.6.1 Создание каталога резервного копирования	15
2.6.2 Создание резервных копий	15
2.6.3 Восстановление базы данных	15
2.7 Перенос конфигурации MariaDB и резервных копий в Vagrant-окружение	15
2.7.1 Подготовка структуры каталогов	15
2.7.2 Создание сценария автоматической установки	16
3 Итоги	17
3.1 Вывод	17
3.2 Контрольные вопросы	17

Список иллюстраций

2.1 Просмотр файлов каталога my.cnf.d	7
2.2 Запуск MariaDB и добавление в автозагрузку	7
2.3 Настройка безопасности MariaDB	8
2.4 Список команд SQL-клиента	9
2.5 Статус MariaDB	10
2.6 Файл конфигурации utf8.cnf	11
2.7 Создание базы данных и проверка её структуры	12
2.8 Создание таблицы и заполнение данными	13
2.9 Создание пользователя, назначение прав и просмотр структуры таблицы	14
2.10 Просмотр баз данных и таблиц	14
2.11 Создание и восстановление резервных копий	15
2.12 Подготовка окружения и создание сценария mysql.sh	16

Список таблиц

1 Введение

1.1 Цель работы

Приобретение практических навыков по установке и конфигурированию системы управления базами данных на примере программного обеспечения MariaDB.

2 Процесс работы

2.1 Установка MariaDB

2.1.1 Установка необходимых пакетов

В систему были установлены пакеты MariaDB-сервера и клиентских утилит. В процессе установки загружены зависимости и сформирована рабочая среда для СУБД.

2.1.2 Просмотр конфигурационных файлов

Конфигурационные файлы MariaDB расположены в каталогах `/etc/my.cnf.d` и `/etc/my.cnf`.

Были просмотрены доступные конфигурации и определены основные директивы.

```

Installed:
  mariadb-3:10.11.11-1.el10.x86_64
  mariadb-client-utils-3:10.11.11-1.el10.x86_64
  mariadb-errmsg-3:10.11.11-1.el10.noarch
  mariadb-server-3:10.11.11-1.el10.x86_64
  mysql-selinux-1.0.14-1.el10_0.noarch
  perl-Sys-Hostname-1.25-512.2.el10_0.x86_64

mariadb-backup-3:10.11.11-1.el10.x86_64
mariadb-common-3:10.11.11-1.el10.noarch
mariadb-gssapi-server-3:10.11.11-1.el10.x86_64
mariadb-server-utils-3:10.11.11-1.el10.x86_64
perl-DBD-MariaDB-1.23-10.el10.x86_64

Complete!
[root@server.dgavdadaev.net ~]# ls /etc/my.cnf.d/
auth_gssapi.cnf  enable_encryption.preset  mysql-clients.cnf  provider_lz4.cnf  provider_snappy.cnf
client.cnf        mariadb-server.cnf      provider_bzip2.cnf  provider_lzo.cnf   spider.cnf
[root@server.dgavdadaev.net ~]# cat /etc/my.cnf
#
# This group is read both by the client and the server
# use it for options that affect everything
#
[client-server]

#
# include all files from the config directory
#
!includedir /etc/my.cnf.d

[root@server.dgavdadaev.net ~]#

```

Рис. 2.1: Просмотр файлов каталога my.cnf.d

Построчные пояснения содержимого основного файла конфигурации:

- Секция предназначена для чтения как клиентом, так и сервером.
- Параметры внутри группы применяются ко всем компонентам MariaDB.
- Определена группа `client-server`, где задаются общие настройки.
- Включается каталог с конфигурациями `/etc/my.cnf.d`, содержащий отдельные `.cnf` файлы.

2.1.3 Запуск MariaDB и включение автозагрузки

После завершения установки MariaDB была запущена и добавлена в автозагрузку операционной системы.

```

[root@server.dgavdadaev.net ~]#
[root@server.dgavdadaev.net ~]# systemctl start mariadb
[root@server.dgavdadaev.net ~]# systemctl enable mariadb
Created symlink '/etc/systemd/system/mysql.service' → '/usr/lib/systemd/system/mariadb.service'.
Created symlink '/etc/systemd/system/mysqld.service' → '/usr/lib/systemd/system/mariadb.service'.
Created symlink '/etc/systemd/system/multi-user.target.wants/mariadb.service' → '/usr/lib/systemd/system/mariadb.service'.

[root@server.dgavdadaev.net ~]# ss -tulpn maria
Error: an inet prefix is expected rather than "maria".
Cannot parse dst/src address.
[root@server.dgavdadaev.net ~]# ss -tulpn | grep maria
tcp  LISTEN  0          0.0.0.0:3306          0.0.0.0:*      users:(("mariadb",pid=17045,fd=18))
                                                uid:27  ino:65760 sk:13 cgroup:/system.s
lice/mariadb.service <->
tcp  LISTEN  0          [::]:3306           [::]:*      users:(("mariadb",pid=17045,fd=19))
                                                uid:27  ino:65761 sk:1f cgroup:/system.s
lice/mariadb.service v6only:1 <->
[root@server.dgavdadaev.net ~]#

```

Рис. 2.2: Запуск MariaDB и добавление в автозагрузку

2.1.4 Проверка работы MariaDB и прослушивания порта

Проверено, что MariaDB корректно слушает порт 3306, что подтверждает успешный запуск сервера.

2.1.5 Настройка безопасности MariaDB

Был выполнен скрипт первичной безопасности.

В ходе выполнения:

- установлен пароль для пользователя root (БД);
- отключён удалённый доступ root;
- удалены анонимные пользователи;
- удалена тестовая база данных;
- обновлены таблицы привилегий.

```
Remove anonymous users? [Y/n]
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n]
... Success!

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n]
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n]
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB
installation should now be secure.

Thanks for using MariaDB!
[root@server.dgavdadaev.net ~]#
```

Рис. 2.3: Настройка безопасности MariaDB

2.1.6 Взаимодействие с MariaDB

Произведён вход в интерактивную оболочку.

Просмотрены доступные клиентские команды и список системных баз данных.

```
List of all client commands:
Note that all text commands must be first on line and end with ';'
?          (\?) Synonym for `help'.
charset   (\C) Switch to another charset. Might be needed for processing binlog with multi-byte charsets.
clear     (\c) Clear the current input statement.
connect   (\r) Reconnect to the server. Optional arguments are db and host.
delimiter (\d) Set statement delimiter.
edit      (\e) Edit command with $EDITOR.
ego       (\G) Send command to MariaDB server, display result vertically.
exit      (\q) Exit mysql. Same as quit.
go        (\g) Send command to MariaDB server.
help      (\h) Display this help.
nopager   (\n) Disable pager, print to stdout.
noteee    (\t) Don't write into outfile.
nowarning (\w) Don't show warnings after every statement.
pager     (\P) Set PAGER [to_pager]. Print the query results via PAGER.
print    (\p) Print current command.
prompt   (\R) Change your mysql prompt.
quit     (\q) Quit mysql.
rehash   (\#) Rebuild completion hash.
sandbox  (\-) Disallow commands that access the file system (except \P without an argument and \e).
source   (\.) Execute an SQL script file. Takes a file name as an argument.
status   (\s) Get status information from the server.
system   (\!) Execute a system shell command.
tee      (\T) Set outfile [to_outfile]. Append everything into given outfile.
use      (\u) Use another database. Takes database name as argument.
warnings (\W) Show warnings after every statement.

For server side help, type 'help contents'

MariaDB [(none)]> SHOW DATABASES;
+-----+
| Database      |
+-----+
| information_schema |
| mysql          |
| performance_schema |
| sys            |
+
```

Рис. 2.4: Список команд SQL-клиента

В системе присутствуют следующие базы данных:

- information_schema
- mysql
- performance_schema
- sys

2.2 Конфигурация кодировки символов

2.2.1 Просмотр статуса MariaDB

Отображён подробный статус MariaDB, включающий:

- идентификатор подключения;
- текущего пользователя;
- параметры SSL;
- используемый сокет;
- версию сервера;
- сведения о кодировках клиента, сервера и соединения;
- время работы сервера и статистические данные.

```
MariaDB [(none)]> status
-----
mysql Ver 15.1 Distrib 10.11.11-MariaDB, for Linux (x86_64) using EditLine wrapper

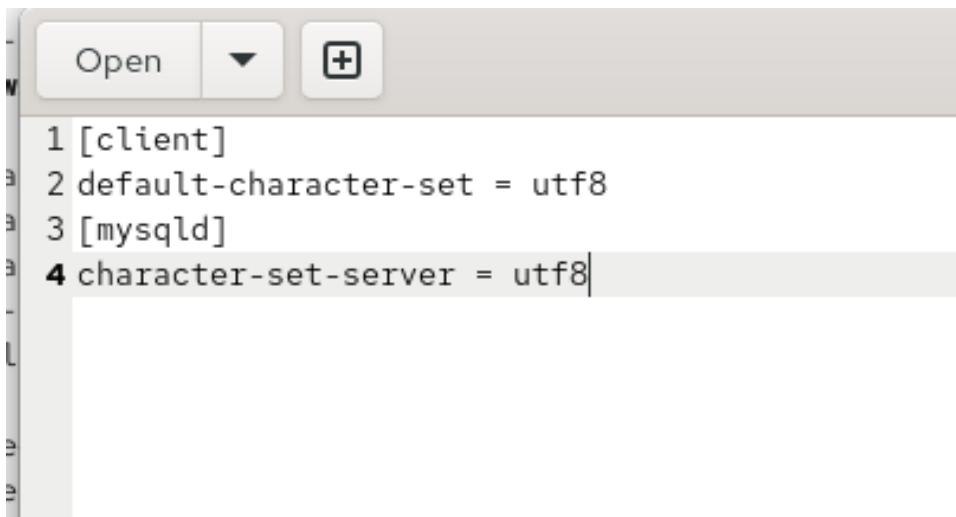
Connection id:          13
Current database:
Current user:           root@localhost
SSL:                   Not in use
Current pager:          stdout
Using outfile:
Using delimiter:         ;
Server:                 MariaDB
Server version:         10.11.11-MariaDB MariaDB Server
Protocol version:        10
Connection:              Localhost via UNIX socket
Server characterset:    latin1
Db     characterset:    latin1
Client characterset:   utf8mb3
Conn.  characterset:   utf8mb3
UNIX socket:            /var/lib/mysql/mysql.sock
Uptime:                 3 min 33 sec

Threads: 1  Questions: 25  Slow queries: 0  Opens: 20  Open tables: 13  Queries per second avg: 0.117
-----
MariaDB [(none)]>
```

Рис. 2.5: Статус MariaDB

2.2.2 Создание конфигурации utf8

В каталоге `/etc/my.cnf.d` создан отдельный конфигурационный файл `utf8.cnf`, задающий кодировку по умолчанию для клиента и сервера.



```
1 [client]
2 default-character-set = utf8
3 [mysqld]
4 character-set-server = utf8
```

Рис. 2.6: Файл конфигурации utf8.cnf

2.2.3 Перезапуск MariaDB и проверка

После перезапуска MariaDB статус сервера подтверждает смену кодировок на utf8, что свидетельствует о корректном применении настройки.

2.3 Создание базы данных addressbook

2.3.1 Создание базы и выбор её в работе

Была создана база данных addressbook с указанием кодировки UTF-8 и выбрана для дальнейшей работы.

После создания база не содержит таблиц, что подтверждает вывод списка доступных таблиц.

```

MariaDB [(none)]> CREATE DATABASE addressbook CHARACTER SET utf8 COLLATE utf8_general_ci;
Query OK, 1 row affected (0.000 sec)

MariaDB [(none)]> USE addressbook;
Database changed
MariaDB [addressbook]> SHOW TABLES;
Empty set (0.000 sec)

MariaDB [addressbook]> CREATE TABLE city(name VARCHAR(40), city Varchar(40));
Query OK, 0 rows affected (0.008 sec)

MariaDB [addressbook]> INSERT INTO city(name, city) Values ('Иванов', 'Москва');
Query OK, 1 row affected (0.004 sec)

MariaDB [addressbook]> INSERT INTO city(name, city) Values ('Петров', 'Сочи');
Query OK, 1 row affected (0.001 sec)

MariaDB [addressbook]> INSERT INTO city(name, city) Values ('Сидоров', 'Дубна');
Query OK, 1 row affected (0.001 sec)

MariaDB [addressbook]> SELECT * FROM city;
+-----+-----+
| name | city |
+-----+-----+
| Иванов | Москва |
| Петров | Сочи |
| Сидоров | Дубна |
+-----+-----+
3 rows in set (0.000 sec)

MariaDB [addressbook]> █

```

Рис. 2.7: Создание базы данных и проверка её структуры

2.3.2 Создание таблицы city

В базе данных создана таблица `city` с двумя текстовыми полями: `name` и `city`. Таблица была заполнена тремя строками: Иванов (Москва), Петров (Сочи), Сидоров (Дубна).

Вывод содержимого таблицы показывает корректное сохранение данных в кодировке UTF-8.

```

MariaDB [addressbook]> CREATE USER dgavdadaev@'%' IDENTIFIED BY '123456';
Query OK, 0 rows affected (0.002 sec)

MariaDB [addressbook]> GRANT SELECT,INSERT,UPDATE,DELETE ON addressbook.* TO dgavdadaev@'%';
Query OK, 0 rows affected (0.002 sec)

MariaDB [addressbook]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.000 sec)

MariaDB [addressbook]> DESCRIBE city;
+-----+-----+-----+-----+
| Field | Type   | Null | Key | Default | Extra |
+-----+-----+-----+-----+
| name  | varchar(40) | YES |     | NULL    |       |
| city   | varchar(40) | YES |     | NULL    |       |
+-----+-----+-----+-----+
2 rows in set (0.001 sec)

MariaDB [addressbook]> 

```

Рис. 2.8: Создание таблицы и заполнение данными

Результат выполнения запроса `SELECT * FROM city` отображает три строки, каждая из которых содержит имя и соответствующий город. Это подтверждает успешное создание таблицы и корректность выполнения операций вставки данных.

2.4 Создание пользователя и настройка его прав

2.4.1 Создание пользователя

Был создан новый пользователь, предназначенный для работы с базой данных `addressbook`, и ему назначен пароль.

2.4.2 Назначение прав

Пользователю предоставлены привилегии на просмотр, добавление, изменение и удаление данных в таблицах базы `addressbook`.

После этого конфигурация прав была обновлена.

Структура таблицы `city` просмотрена с помощью команды `DESCRIBE` — она содержит два столбца `name` и `city` типа `VARCHAR(40)`.

```
[root@server.dgavdadaev.net my.cnf.d]#
[root@server.dgavdadaev.net my.cnf.d]# mysqlshow -u root -p
Enter password:
+-----+
|   Databases   |
+-----+
| addressbook   |
| information_schema |
| mysql          |
| performance_schema |
| sys            |
+-----+
[root@server.dgavdadaev.net my.cnf.d]#
[root@server.dgavdadaev.net my.cnf.d]# mysqlshow -u dgavdadaev -p addressbook
Enter password:
Database: addressbook
+-----+
| Tables |
+-----+
| city   |
+-----+
[root@server.dgavdadaev.net my.cnf.d]# █
```

Рис. 2.9: Создание пользователя, назначение прав и просмотр структуры таблицы

2.5 Просмотр баз данных и таблиц

Список всех доступных баз данных был выведен с учётной записью администратора.

Затем был просмотрен перечень таблиц внутри базы addressbook как под пользователем root, так и под созданным пользователем.

```
[root@server.dgavdadaev.net my.cnf.d]#
[root@server.dgavdadaev.net my.cnf.d]# mkdir -p /var/backup
[root@server.dgavdadaev.net my.cnf.d]# mysqldump -u root -p addressbook > /var/backup/addressbook.sql
Enter password:
[root@server.dgavdadaev.net my.cnf.d]# cd /var/backup/
[root@server.dgavdadaev.net backup]# mysqldump -u root -p addressbook | gzip > addressbook.sql.gz
Enter password:
[root@server.dgavdadaev.net backup]# mysqldump -u root -p addressbook | gzip > $(date +addressbook.%Y%m%d.%H%M%S.sql.gz)
Enter password:
[root@server.dgavdadaev.net backup]# ls
addressbook.20251128.094104.sql.gz addressbook.sql addressbook.sql.gz
[root@server.dgavdadaev.net backup]# mysql -u root -p addressbook < addressbook.sql
Enter password:
[root@server.dgavdadaev.net backup]# zcat addressbook.sql.gz | mysql -u root -p addressbook
Enter password:
[root@server.dgavdadaev.net backup]#
```

Рис. 2.10: Просмотр баз данных и таблиц

2.6 Работа с резервными копиями

2.6.1 Создание каталога резервного копирования

На виртуальной машине создан каталог `/var/backup`, предназначенный для хранения копий баз данных.

2.6.2 Создание резервных копий

Были выполнены следующие операции:

- создана обычная резервная копия базы `addressbook`;
- создана сжатая копия;
- создана сжатая копия с указанием даты и времени создания.

2.6.3 Восстановление базы данных

База данных была успешно восстановлена как из несжатой копии, так и из сжатой версии при помощи инструмента `zcat`.

```
[root@server.dgavdadaev.net backup]#  
[root@server.dgavdadaev.net backup]# cd /vagrant/provision/server/  
[root@server.dgavdadaev.net server]# mkdir -p /vagrant/provision/server/mysql/etc/my.cnf.d  
[root@server.dgavdadaev.net server]# mkdir -p /vagrant/provision/server/mysql/var/backup  
[root@server.dgavdadaev.net server]# cp -R /etc/my.cnf.d/utf8.cnf /vagrant/provision/server/mysql/etc/my.cnf.d  
[root@server.dgavdadaev.net server]# cp -R /var/backup/* /vagrant/provision/server/mysql/var/backup  
[root@server.dgavdadaev.net server]# touch mysql.sh  
[root@server.dgavdadaev.net server]#
```

Рис. 2.11: Создание и восстановление резервных копий

2.7 Перенос конфигурации MariaDB и резервных копий в Vagrant-окружение

2.7.1 Подготовка структуры каталогов

В каталоге `/vagrant/provision/server` был создан новый каталог `mysql` с подкаталогами для конфигурационных файлов MariaDB и резервных копий.

В соответствующие каталоги были скопированы:

- файл `utf8.cnf` из системного каталога MariaDB,
- резервные копии базы данных из `/var/backup`.

2.7.2 Создание сценария автоматической установки

Создан исполняемый файл `mysql.sh`, предназначенный для автоматизации установки MariaDB, копирования конфигураций, выполнения начальной настройки и развертывания базы данных из резервной копии.

```
1  #!/bin/bash
2  echo "Provisioning script $0"
3  systemctl restart named
4  echo "Install needed packages"
5  dnf -y install mariadb mariadb-server
6  echo "Copy configuration files"
7  cp -R /vagrant/provision/server/mysql/etc/* /etc
8  mkdir -p /var/backup
9  cp -R /vagrant/provision/server/mysql/var/backup/* /var/backup
10 echo "Start mysql service"
11 systemctl enable mariadb
12 systemctl start mariadb
13 if [[ ! -d /var/lib/mysql/mysql ]]
14 then
15   echo "Securing mariadb"
16   mysql_secure_installation <<EOF
17   y
18   123456
19   123456
20   y
21   Y
22   y
23   y
24   EOF
25   echo "Create database"
26   mysql -u root -p123456 <<EOF
27   CREATE DATABASE addressbook CHARACTER SET utf8 COLLATE utf8_general_ci;
28   EOF
29   mysql -u root -p123456 addressbook < /var/backup/addressbook.sql
30 fi
```

Рис. 2.12: Подготовка окружения и создание сценария `mysql.sh`

3 Итоги

3.1 Вывод

В ходе выполнения работы была создана база данных `addressbook`, сформирована таблица `city` и заполнена данными. Настроен пользователь с необходимыми правами доступа, выполнены операции резервного копирования и восстановления базы. Конфигурационные файлы и резервные копии были интегрированы в среду Vagrant для автоматизации последующих развёртываний. Все задания выполнены успешно, а результаты подтверждены соответствующими выводами MariaDB.

3.2 Контрольные вопросы

1. Какая команда отвечает за настройки безопасности в MariaDB?

Команда `mysql_secure_installation` выполняет начальную настройку безопасности: установку пароля `root`, удаление анонимных пользователей, отключение удалённого доступа и удаление тестовой базы.

2. Как настроить MariaDB для доступа через сеть?

Необходимо изменить параметр `bind-address` в конфигурации MariaDB, задав IP-адрес, к которому разрешено подключение, либо значение `0.0.0.0` для приёма подключений со всех сетей. После изменения требуется перезапуск сервиса.

3. Какая команда позволяет получить обзор доступных баз данных после входа в среду оболочки MariaDB?

Команда SHOW DATABASES выводит список всех доступных баз данных.

4. Какая команда позволяет узнать, какие таблицы доступны в базе данных?

Команда SHOW TABLES выводит перечень таблиц текущей выбранной базы данных.

5. Какая команда позволяет узнать, какие поля доступны в таблице?

Команда DESCRIBE table_name отображает структуру таблицы, имена полей и их типы.

6. Какая команда позволяет узнать, какие записи доступны в таблице?

Команда SELECT * FROM table_name выводит все записи таблицы.

7. Как удалить запись из таблицы?

Удаление выполняется через команду DELETE FROM table_name WHERE условие, где условие определяет удаляемую строку.

8. Где расположены файлы конфигурации MariaDB? Что можно настроить с их помощью?

Основные конфигурационные файлы находятся в /etc/my.cnf и каталоге /etc/my.cnf.d/.

Через них можно задавать параметры сервера и клиента: кодировку, сетевые настройки, каталоги хранения данных, параметры безопасности, подключение внешних файлов конфигураций.

9. Где располагаются файлы с базами данных MariaDB?

Файлы баз данных находятся в каталоге /var/lib/mysql, где каждая база представлена отдельным подкаталогом.

10. Как сделать резервную копию базы данных и затем её восстановить?

Резервная копия создаётся утилитой mysqldump, а восстановление проводится импортом SQL-файла в MariaDB. Возможны как обычные, так и сжатые варианты хранения копий.